

In the Claims:

Please amend claim 34 as follows:

1. (Previous presented) A method for providing parallel execution of computing tasks in a heterogeneous computing environment comprising:
 - partitioning a computing task into small tasks;
 - assigning the small tasks to mobile agents;
 - determining available computing hosts in the heterogeneous computing environment;
 - transferring the mobile agents to the available computing hosts;
 - executing the mobile agents at the available computing hosts using execution code provided to the mobile agents from a central server; and
 - maintaining, at a second computing host at which none of the mobile agents are executing, stack trace and state information about each of the mobile agents to allow one or more of the mobile agents to be reconstructed at an alternate computing host using the stack trace and state information.
2. (Original) The method of claim 1, wherein the step of determining available computing hosts further comprises determining network latency.
3. (Previous presented) The method of claim 1, further comprising transferring execution code from a the central server to virtual machines at the available computing hosts
4. (Original) The method of claim 1, wherein the step of executing the mobile agents is performed in virtual machines at the computing hosts without modification of the virtual machines.
5. (Original) The method of claim 1, wherein one of the small tasks is assigned to one of the mobile agents.
6. (Original) The method of claim 3, wherein the step of transferring execution code is performed in response to a request by a mobile agent.
7. (Original) The method of claim 6, wherein the step of transferring execution code is performed by a web server.

8. (Original) The method of claim 1 further comprising monitoring execution of the mobile agents at the available computing hosts.

9. (Previous presented) The method of claim 1 further comprising allowing the mobile agents to collaborate over the heterogeneous computing environment as the mobile agents execute at the available computing hosts.

10. (Original) The method of claim 1, wherein the mobile agents are executed in real time.

11. (Previous presented) A method for providing parallel computing using mobile agents comprising:

assigning a computing task to one or more mobile agents;

transferring the one or more mobile agents to one or more available computing hosts;

transferring execution code from a central server to the one or more available computing hosts;

executing the one or more mobile agents at the one or more available computing hosts using the execution code; and

storing stack trace and real-time state information about the one or more mobile agents at a first alternate computing host on which none of the one or more mobile agents are executing to allow the one or more mobile agents to be reconstructed at a second alternate computing host.

12. (Original) The method of claim 11 further comprising, prior to transferring the mobile agents, the steps of:

determining network latency affecting transmission of data between computing hosts; and

halting transferring of the mobile agents if network latency exceeds a threshold.

13. (Original) The method of claim 12 further comprising, prior to transferring the mobile agents, the steps of:

monitoring utilization of the computing hosts; and

halting transferring of the mobile agents if utilization of computing hosts exceeds a threshold.

14. (Previous presented) A method for migrating a software application running in a virtual machine from a primary host to a secondary host comprising:

constructing an application using a plurality of mobile agents;
transferring the plurality of mobile agents to a first computing host;
executing the plurality of mobile agents at the first computing host;
maintaining stack trace and state information about each of the plurality of mobile agents at a second computing host on which none of the plurality of mobile agents are executing;

detecting an indication to migrate the application; and
in response to the indication, migrating the application in its entirety from the first computing host to a third computing host without modifying a virtual machine at the third computing host by reconstructing each of the plurality of mobile agents at the third computing host using the stack trace and state information.

15. (Original) The method of claim 14 wherein the indication comprises an indication of network latency.

16. (Original) The method of claim 14 wherein the indication comprises an indication of hostile attack.

17. (Original) The method of claim 16 wherein the hostile attack comprises hacking.

18. (Original) The method of claim 14 wherein the indication comprises an indication of network failure.

19. (Original) The method of claim 14 wherein the indication comprises an indication of computer hardware failure.

20. (Previous presented) The method of claim 14 further comprising resuming execution of the mobile agent at the third computing host at a point where execution was halted.

21. (Previous presented) The method of claim 20, wherein the stack trace and state information comprises information about an execution thread of the mobile agent as it existed at the first host prior to being transferred to the third computing host.

22. (Cancelled)

23. (Cancelled)

24. (Cancelled)

25. (Original) The method of claim 14 further comprising:
continuing monitoring for another indication to migrate the application;
continuing migrating the application to other hosts.

26. (Previous presented) A computer system apparatus for providing parallel execution of computing tasks in a heterogeneous computing environment comprising:
a dispatcher for partitioning the computing task into a plurality of small tasks and dispatching the small tasks;
mobile agents for receiving small tasks from the dispatcher;
computing resources on a network including virtual machines for executing mobile agent software code;
means for transferring the mobile agents to the computing resources; and
means for transferring execution code from a central server to the computing resources, the computing resources receiving and executing one of the small tasks assigned to a mobile agent in the virtual machines using the execution code and the means for transferring execution code maintaining stack trace and state information about each of the mobile agents at a first alternate computing host where none of the mobile agents are executing to allow each of the mobile agents to be reconstructed at a second alternate computing host.

27. (Cancelled)

28. (Original) The apparatus of claim 26 wherein the central server comprises a web server.

29. (Original) The apparatus of claim 26 further comprising means for monitoring execution of the small tasks.

30. (Original) The apparatus of claim 26 further comprising collaboration means for allowing the mobile agents to communicate and share information in real time.

31. (Original) The apparatus of claim 26, wherein the mobile agents execute in real time.

32. (Original) The apparatus of claim 26, further comprising storage means for storing real time state information about the mobile agents as the mobile agents execute at the computing resources.

33. (Original) The apparatus of claim 26 further comprising:
means for monitoring execution of the mobile agents at the computing hosts; and
means for detecting over-utilization of one of the computing hosts and for issuing a warning when one of the computing hosts is over-utilized.

34. (Currently amended) A method for providing realistic thread migration which comprises:

instantiating a mobile agent thread at a first computing host;
processing the mobile agent thread at the first computing host;
storing stack trace and state information about the mobile agent thread at a second computing host at which the mobile agent thread is not executing as the mobile agent thread executes at the first computing host;
detecting an indication to migrate the mobile agent thread; and
in response to the indication,
stopping execution of the mobile agent thread;
transferring the execution code for the mobile agent thread from a central server to a third computing host; and
transferring the stack trace and state information about the mobile agent thread to a the third computing host.

35. (Cancelled)

36. (Cancelled)

37. (Cancelled)

38. (Previous presented) The method of claim 34 further comprising:
receiving the stack trace and state information about the mobile agent thread at the third computing host;
reconstructing the mobile agent thread at the third computing host using the stack trace and state information about the mobile agent thread; and
continuing processing of the mobile agent thread at the third computing host at a point at which execution of the thread was stopped at the first computing host.

39. (Previous presented) The method of claim 38 wherein the step of transferring the stack trace and state information about the mobile agent thread further comprises serializing the information about the mobile agent thread.

40. (Previous presented) An agent collaboration environment which comprises:

a plurality of mobile agents;
an agent debugger for storing stack trace and state information about each of the plurality of mobile agents at a computing host where none of the plurality of mobile agents are executing;
a conference room for providing a virtual workspace for the mobile agents; and
a registration subsystem for selectively assigning the plurality of mobile agents to the conference room,
wherein each of the plurality of agents can share data, information, and results of computations in the conference room.

41. (Original) The agent collaboration environment of claim 40, wherein the conference room monitors and moderates communication between the plurality of mobile agents.

42. (Previous presented) A method for dynamically constructing and executing a computer application in a heterogeneous computing environment using mobile agents comprising:

constructing a computer application using mobile agents;
storing the mobile agents at a central server;

dispatching the mobile agents from the central server to an available computing host on the heterogeneous computing environment using a dispatching component;
allowing the mobile agents to execute at the available computing host; and
monitoring execution of each of the plurality of mobile agents and storing stack trace and state information about execution of the mobile agents at a computing host where none of the mobile agents are executing.

43. (Previous presented) The method of claim 42, further comprising monitoring CPU utilization and network latency prior to dispatching the mobile agents.

44. (Previous presented) The method of claim 42, further comprising providing a collaboration environment for allowing the mobile agents to exchange information in real time.

45. (Previous presented) The method of claim 42, wherein the step of allowing the mobile agents to execute at the available computing hosts comprises allowing the mobile agents to execute at the available computing host without requiring prior installation of executable code at the available computing host.

46. (Previous presented) The method of claim 1, further comprising reconstructing one or more of the mobile agents at the second computing host using the stack trace and state information stored at the first computing host.

47. (Previous presented) The method of claim 11, wherein the step of transferring the one or more mobile agents to one or more available computing hosts comprises transferring data relating to the one or more mobile agents to the one or more mobile computing hosts.

48. (Previous presented) The method of claim 26, wherein the means for transferring the mobile agents to the computing resources comprises means for transferring data relating to the mobile agents to the computing resources.

49. (Previous presented) The apparatus of claim 28, wherein the web server is located at a first organization.

50. (Previous presented) The apparatus of claim 49, further comprising an agent execution environment located at a second organization.

51. (Previous presented) The apparatus of claim 50, further comprising firewalls at the first and second organizations to allow execution code for the mobile agents to be exchanged between the first and second organizations.